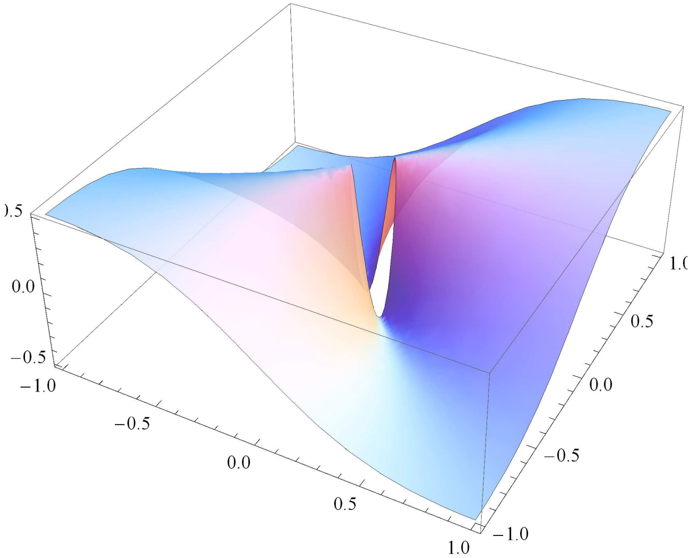
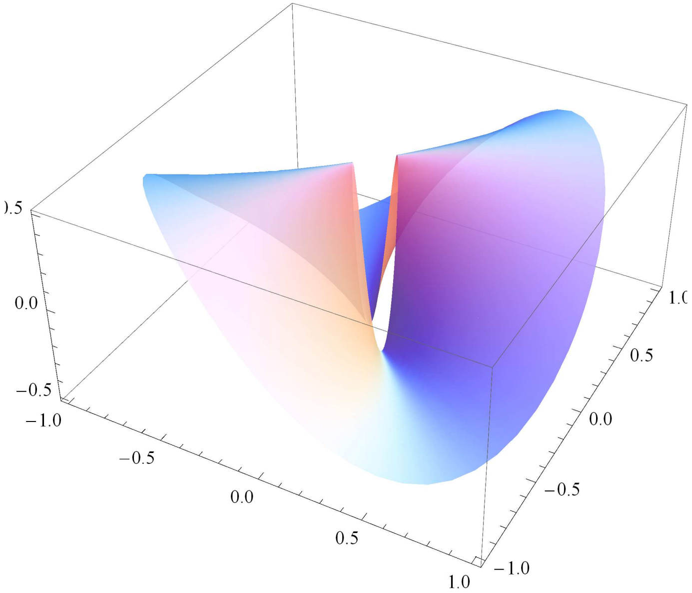
```
Clear["Global`*"];
Plot3D[x*y/(x^2 + y^2), {x, -1, 1}, {y, -1, 1}, Mesh → None, RegionFunction -> Function[{x, y, z}, x^2 + y^2 > 0.01], PlotStyle → Opacity[0.9]]
```



```
ParametricPlot3D[{r Cos[θ], r Sin[θ], 1/2 Sin[2 θ]}, {r, 0.1, 1}, {θ, 0, 2 π}, Mesh → None, PlotPoints → {20, 60}, PlotStyle → Opacity[0.9]]
```




```
RealToHolo[expr_, anum_, {xsym_, ysym_, zsym_ }] :=
 Module[{abar = Conjugate[anum], exprf},
  exprf = ComplexExpand[expr, TargetFunctions → {Re, Im}];
  func =
   2 * exprf /. {xsym → (zsym + abar) / 2, ysym → (zsym - abar) / (2 * I)}; basecorr = - exprf /. {xsym → Re[anum], ysym → Im[anum]};
  FullSimplify[func + basecorr + I * β]]
```

```
RealToHolo[4 x * y (y^2 - x^2), 0, {x, y, z }]
```

$i\,(z^4 + \beta)$

```
laplacian[expr_, {xsym_, ysym_}] :=
  Module[{laplaciano}, laplaciano = FullSimplify[Together[D[expr, {xsym, 2}] + D[expr, {ysym, 2}]]]; laplaciano];
```

laplacian$\left[\frac{1}{2}\text{Log}\left[x^2+y^2\right], \{x, y\}\right]$

```
0
```

```
HarmonicConjugate[expr_, anum_, {xsym_, ysym_}] :=
  Module[{abar = Conjugate[anum], zsym, exprf}, exprf = ComplexExpand[expr, TargetFunctions → {Re, Im}];
   func = 2 * exprf /. {xsym → (zsym + abar) / 2, ysym → (zsym - abar) / (2 * I)}; basecorr = -exprf /. {xsym → Re[anum], ysym → Im[anum]};
   ComplexExpand[Im[FullSimplify[(func + basecorr + I * β) /. zsym → xsym + I * ysym]]]];
```

HarmonicConjugate$\left[\frac{1}{2}\text{Log}\left[x^2+y^2\right], 1, \{x, y\}\right]$

$\beta + \text{Arg}[x + i\, y]$

```
HarmonicConjugate[(x^2 + y^2)^(1/4) Cos[(1/2) ArcTan[x, y]], 1, {x, y}]
```

$\beta + \left(x^2+y^2\right)^{1/4}\text{Sin}\left[\frac{1}{2}\text{Arg}[x + i\, y]\right]$

```
? Limit
```

Limit[*expr*, *x* –> x_0] finds the limiting value of *expr* when *x* approaches x_0. ≫

```
? D
```

D[*f*, *x*] gives the partial derivative $\partial f / \partial x$.
D[*f*, {*x*, *n*}] gives the multiple derivative $\partial^n f / \partial x^n$.
D[*f*, *x*, *y*, …] differentiates f successively with respect to x, y, ….
D[*f*, {{x_1, x_2, …}}] for a scalar f gives the vector derivative $(\partial f / \partial x_1, \partial f / \partial x_2, \ldots)$. ≫